

REMARKS

Claims 1-8, 10-20, and 22-31 are pending in the Application.

Claims 1-8, 10-20, and 22-31 stand rejected.

Claims 1-6, 8, 11, 20, 24, and 25 have been amended. No new matter has been added. Support for the amendments to Claims 1, 8, 11, 20, 24, and 25, can be found, at least, within paragraphs [0010]-[0016]. Amendments to Claims 2-6 have been made for consistency on the independent claims on which they respectively depend.

Examiner Interview

Appreciation is expressed for the Examiner interview conducted on 10/21/2009 with Examiner Tank. During the interview, the cited references were discussed with regard to independent claim 1. While no agreement was reached during the interview, the Examiner provided insight regarding which elements of the originally-filed Specification may serve to overcome the currently cited references. The undersigned believes the amendments presented in this paper are in harmony with the positions expressed by the Examiner during the interview.

Rejection of Claims under 35 U.S.C. §103

Claims 1-8, 10-19, and 25-31

Claims 1-8, 10-19, and 25-31 stand rejected under 35 U.S.C. § 103(a) as purportedly being unpatentable over U.S. Patent No. 5,831,609, issued to London et al. (“London”), in view of Hallberg et al, “Using Microsoft Excel 97” (“Hallberg”). Applicants respectfully traverse this rejection.

Amended independent Claim 1:

providing information relating to an external business application in a server system,
comprising
receiving a request from an internal business application, wherein
the request comprises
an execute element, and
an argument element,
the execute element is configured to cause the external business
application to execute a command of the external business
application, and
the argument element comprises an indication of one or more user
interface elements that are to be returned;
generating a data element by causing the business application to execute the
command;
generating the one or more user interface elements, wherein
the one or more user interface elements correspond to a subset of user
interface elements provided by the external business application,
and
the subset is selected according to the argument element; and
sending a response to the internal business application, comprising the one or
more user interface elements and the data element.

(Emphasis added).

Independent Claims 8, 11, 20, 24, and 25 recite a similar “generating” limitation. Applicants respectfully submit that neither London nor Hallberg, alone or in any combination, teach or suggest in any way, at least: (1) generating one or more user interface elements corresponding to a subset of user interface elements provided by an external business application; and (2) that the subset is selected according to an argument element in a request from an internal business application. No teaching of these claimed elements is possible because neither reference provides any means by which user interface elements are selectively chosen to be returned from one business application to another business application. Without any selection

of user interface elements, an ordinary artisan would be unable to extend the cited references to anything like the claimed invention.

London pertains broadly to translation software that allows a user on a local terminal, running one operating system, to view an application running on a remote terminal, running a different operating system. London purportedly translates the GUI instructions on the remote terminal into instructions compatible with the local terminal. *See* London, Abstract.

As an initial matter, Applicants note that London is not concerned with a finer granularity than the entire GUI interface presented on the remote terminal. In other words, London's translation is intended to allow a user a local terminal to see what the user would see if the user were at the remote terminal. The user sees, in effect, the image of the GUI interface on the local terminal. There is no indication in the cited sections of London (or elsewhere) that any selection of GUI elements to be translated is performed. In other words, in London, the entire GUI is always translated, and nothing less.

The Office Actions cites the following section of London as purported disclosure of "generating the user interface element to be returned in response to the request, wherein the user interface element is configured according to the command block" (from a previous version of Claim 1):

The CreateWindow API call creates the desired window internally in the memory of the computer system. To display the window on the video display, the program issues a ShowWindow API call (step 420). A processing loop, collectively represented by blocks 430, 435, 440 and 450, then requests the painting of the text "Hello" on the video display. The message loop 430 refers to the code that retrieves messages from a message queue provided by the native operating system. More specifically, when the program receives a paint message (step 435, YES PATHWAY), the program requests the rendering of the character string "Hello" in the center of the displayed window (step 440).

London 6:17-27.

Applicants respectfully submit that this section of London does not disclose any operation even remotely comparable to the claimed selection of user interface elements. Much less, this section of London does not disclose that one or more user interface elements correspond to a subset of user interface elements provided by an external business application, where the subset is selected according to an argument element in a request from an internal business application. This section of London is only describing the process by which London displays the string "Hello" on the local terminal when the string "Hello" is displayed on the

remote terminal. There is nothing in this section of London to indicate that any aspect of the user interface that is displayed on the remote terminal is not displayed on the local terminal. In other words, there is no selection of user interface elements in London's translation method.

By contrast, the claimed invention is not directed to a solution such as that of London. The claimed method allows for an internal business application to integrate the output of an external business application. Such integration permits the internal business application to seamlessly present, in a consistent user interface, the output data generated by the external business application. This consistency in the user interface is possible because the internal business application can specify, in a request made to the external business application, which subset of user interface elements provided by the external business application are to be selected for use by the internal business application.

Given London, an ordinary artisan would not find any teachings of decomposing the user interface elements in the GUI processed by London's method. Further, an ordinary artisan would not even consider such an incongruous concept when contemplating London. This is at least because the cited sections of London disclose no purpose remotely similar to the integrating of only a selected subset of user interface elements of the output of one business application into another business application. Without such a purpose, an ordinary artisan would not even consider decomposing a GUI into user interface elements because this would only add an unnecessary layer of complication to London's method, without providing any benefit.

Hallberg, correctly, is not cited as disclosing anything pertaining to the "generating" limitation and therefore Hallberg is clearly unable to cure this deficiency of London, among others. Thus, neither Hallberg nor London, alone or in any combination, teach or suggest all of the elements of the claimed invention.

For at least these reasons, Applicants submit that neither London nor Hallberg, alone or in combination, provide disclosure of all the limitations of independent claims 1, 8, 11, 20, 24, and 25, and all claims depending therefrom, and that these claims are in condition for allowance. Applicants therefore respectfully request the Examiner's reconsideration and withdrawal of the rejections to these claims and an indication of the allowability of same.

Claims 20, 22, and 23

Claims 20 and 22-23 are rejected under 35 U.S.C. 103(a) as purportedly being unpatentable over U.S. Patent No. 5,761,673 B1, issued to Bookman et al. (“Bookman”).

Applicants respectfully traverse this rejection.

Amended independent Claim 20:

first instructions, executable on a first computer system, configured to execute a first command of a first business application, wherein the first command is represented by a first command block;
second instructions, executable on a second computer system, configured to execute a second command of a second business application, wherein the second command is represented by a second command block; and
a common data structure defining the first command block and the second command block, wherein the first command block and the second command block are inbound to a web server, and
the common data structure comprises
 an execute element having a path attribute indicating a location of an object manager,
 a command element nested within the execute element comprising a value attribute indicating a name of a command, and
 one or more argument elements nested within the command element, wherein
 each argument element comprises a name attribute indicating a name of an argument for the command,
 the one or more argument elements being from a set of argument elements comprising an argument element configured to indicate a response markup format,
 an argument element configured to indicate whether the response should include user interface elements,
 select, when the argument element indicates the response should not comprise user interface elements, an empty set of user interface elements,
 select, when the argument element indicates the response should comprise user interface elements, a subset of user interface elements according to the argument element, and
identify a transform to be applied to output.

(Emphasis added).

Applicants respectfully submit that Bookman fails to teach or suggest in any way, at least: (1) selecting either a subset or an empty set of user interface elements; and (2) that the subset or empty set is selected according to an argument element in a request from an internal business application. No teaching of these claimed elements is possible because Bookman does not provide any means by which user interface elements are selectively chosen to be returned from one business application to another business application. Without any selection of user interface elements, an ordinary artisan would be unable to extend Bookman to comprehend anything like the claimed invention.

Bookman pertains broadly to generating dynamic web pages. *See* Bookman, Abstract. Bookman purportedly accomplishes the dynamic generation of web pages by invoking and executing predefined procedural packages stored in a database. In other words, Bookman's method generates a complete web page from the data generated by executing a predefined procedure stored on a database. However, there is no indication that any aspects of the user interface elements from Bookman's generated web pages are treated as less than a whole web page. This is to say that while Bookman may generate a web page, Bookman does not envision that anything less than all of the elements of the user interface of the web page will be presented to the requesting client. Given that Bookman provides no indication whatsoever that either a subset or empty subset of user interface elements in the generated web page might somehow be selected, it cannot be said that Bookman teaches or contemplates the claimed limitations.

The Office Action cites the following section of Bookman as purported disclosure of "an argument element configured to indicate whether the response should include user interface elements, and an argument[] element configured to identify a transform to be applied to output" (from a previous version of Claim 20):

If the object is an HTML file, a GIF file or other files supported by Web browsers, Web server executable 202 retrieves and returns object 205(1) or 207(1) to Web Browser 201. If the object requested is a CGI script, however, CGI script 203 will generate dynamically created HTML output and transfer the output to Web server executable 202, and back to requesting Web browser 201.

Bookman, 4:7-13.

Applicants respectfully submit that this section of Bookman pertains to what type of output is generated and returned to the requesting web browser depending on the object type. That is, if Bookman's object is an HTML file or a GIF file, then the web server retrieves and

returns object 205(1) or 207(1) to the web browser. Bookman's objects 205(1) are files in the operating system on the web server and objects 207(1) are files in the database. Further, according to the above-cited section, if the object requested is a CGI script, what is returned is a dynamically created HTML output. In other words, this section discusses what type of output is generated by Bookman's system based on certain conditions. However, this case-by-case presentation in Bookman on how to generate an output is not remotely similar to the claimed integrating of only a selected subset of user interface elements of the output of one business application into another business application. In short, this section of Bookman is completely silent on considering the output to be returned as anything like a collection of user interface elements – of which only a subset will be selected to be returned to a requesting business application.

By contrast, the claimed invention allows an internal business application to integrate the output of an external business application. Such integration permits the internal business application to seamlessly present, in a consistent user interface, the output data generated by the external business application. This consistency in the user interface is possible because the internal business application can specify, in a request made to the external business application, which subset of user interface elements provided by the external business application are to be selected for use by the internal business application.

Given Bookman, an ordinary artisan would not consider decomposing the user interface elements handled by Bookman's method. This is at least because the cited sections of Bookman disclose no purpose remotely similar to the integrating of only a selected subset of user interface elements of the output of one business application into another business application. Without such a purpose, an ordinary artisan would not decompose a GUI into user interface elements because this would only add an unnecessary layer of complication to Bookman's method, without providing any benefit.

For at least these reasons, Applicants submit that Bookman in light of the skill of an ordinary artisan, fail to provide disclosure of all the limitations of independent claims 20, 22, and 23, and that these claims are in condition for allowance. Applicants therefore respectfully request the Examiner's reconsideration and withdrawal of the rejections to these claims and an indication of the allowability of same.

Claim 24

Claim 24 is rejected under 35 U.S.C. 103(a) as purportedly being unpatentable over Bookman in view of Hallberg. Applicants respectfully traverse this rejection.

Claim 24 recites a similar “subset” limitation as in independent Claims 20. Thus, for reasons similar to those presented above, the “subset” limitation of Claim 20 is not taught or suggested by any combination of Bookman or Hallberg.

For at least these reasons, Applicants submit that neither Bookman nor Hallberg, alone or in combination, provide disclosure of all the limitations of independent Claim 24, and that this claim is in condition for allowance. Applicants therefore respectfully request the Examiner’s reconsideration and withdrawal of the rejections to this claim and an indication of the allowability of same.

CONCLUSION

Applicants submit that all claims are now in condition for allowance, and an early notice to that effect is earnestly solicited. Nonetheless, should any issues remain that might be subject to resolution through a telephonic interview, the Examiner is requested to telephone the undersigned.

If any extensions of time under 37 C.F.R. § 1.136(a) are required in order for this submission to be considered timely, Applicants hereby petition for such extensions. Applicants also hereby authorize that any fees due for such extensions or any other fee associated with this submission, as specified in 37 C.F.R. § 1.16 or § 1.17, be charged to deposit account 502306.

Respectfully submitted,

/ Samuel G. Campbell III /

Samuel G. Campbell III
Attorney for Applicants
Reg. No. 42,381
Telephone: (512) 439-5084
Facsimile: (512) 439-5099